2001年秋天互联网公司（dot-com)泡沫的破灭标志着互联网的一个转折点。许多人断定互联网被过分炒作，事实上网络泡沫和相继而来的股市大衰退看起来像是所有技术革命的共同特征。股市大衰退通常标志着蒸蒸日上的技术已经开始占领中央舞台。假冒者被驱逐，而真正成功的故事展示了它们的力量，同时人们开始理解了是什么将一个故事同另外一个区分开来。   
  
　　“Web 2.0”的概念开始于一个会议中，展开于O'Reilly公司和MediaLive国际公司之间的头脑风暴部分。所谓互联网先驱和O'Reilly公司副总裁的戴尔·多尔蒂(Dale Dougherty)注意到，同所谓的“崩溃”迥然不同，互联网比其他任何时候都更重要，令人激动的新应用程序和网站正在以令人惊讶的规律性涌现出来。更重要的是，那些幸免于当初网络泡沫的公司，看起来有一些共同之处。那么会不会是互联网公司那场泡沫的破灭标志了互联网的一种转折，以至于呼吁“Web 2.0”的行动有了意义？我们都认同这种观点，Web 2.0会议由此诞生。   
  
　　在那个会议之后的一年半的时间里，“Web 2.0”一词已经深入人心，从Google上可以搜索到950万以上的链接。但是，至今关于Web 2.0的含义仍存在极大的分歧，一些人将Web 2.0贬低为毫无疑义的一个行销炒作口号，而其他一些人则将之理解为一种新的传统理念。   
  
　　本文就是来尝试澄清Web 2.0本来意义。   
　　   
　　在我们当初的头脑风暴中，我们已经用一些例子，公式化地表达了我们对Web 2.0的理解：

|  |  |
| --- | --- |
| Web 1.0 | Web 2.0 |
| DoubleClick | Google AdSense |
| Ofoto | Flickr |
| Akamai | BitTorrent |
| mp3.com | Napster |
| 大英百科全书在线（Britannica Online） | 维基百科全书（Wikipedia） |
| 个人网站 | 博客(blogging) |
| evite | upcoming.org和EVDB |
| 域名投机 | 搜索引擎优化 |
| 页面浏览数 | 每次点击成本 |
| 屏幕抓取（screen scraping） | 网络服务（web services） |
| 发布 | 参与 |
| 内容管理系统 | 维基 |
| 目录（分类) | 标签（“分众分类”，folksonomy） |
| 粘性 | 聚合 |

　　这个列表还会不断继续下去。但是到底是什么，使得我们认定一个应用程序或一种方式为作所谓“Web 1.0”，而把另外一个叫做“Web 2.0”呢？（这个问题尤为紧迫，因为Web 2.0的观念已经传播的如此广泛，以至于很多公司正在将这个词加到他们的行销炒作中，但却没有真正理解其含义。同时这个问题也尤为困难，因为许多嗜好口号的创业公司显然不是Web 2.0，而一些我们认为是Web 2.0的应用程序，例如Napster和BitTorrent，甚至不是真正适当的网络程序！）我们首先来探讨一些原则，这些原则是通过Web 1.0的一些成功案例，以及一些最为有趣的新型应用程序来体现的。   
　　   
  
　　1. 互联网作为平台   
  
　　正如许多重要的理念一样，Web 2.0没有一个明确的界限，而是一个重力核心。不妨将Web 2.0视作一组原则和实践，由此来把距离核心或远或近的网站组成为一个类似太阳系的网络系统，这些网站或多或少地体现着Web 2.0的原则。

图1为Web 2.0的“模拟图”，该图是在名为“O'Reilly的朋友”（Friend Of O’reilly, FOO）的会议的一个研讨会上产生的。这个图基本上仍处于演化阶段，但已经描绘出了 从Web 2.0核心理念中衍生出的许多概念。   
  
　　例如，在2004年10月的第一次Web 2.0的会议上，约翰·巴特利（John Battelle）和我在我们各自的开场白中列举了一组初步的原则。   
  
　　这些原则中的第一条就是“互联网作为平台”。这也曾是Web 1.0的宠儿网景公司（Netscape)的战斗口号，而网景在同微软的大战中陨落了。此外，我们早先的Web 1.0的楷模中的两个，DoubleClick和Akamai公司，皆是将网络当作平台的先驱。人们往往不认为这是一种网络服务，但事实上，广告服务是第一个被广泛应用的网络服务，同时也是第一个被广泛应用的混合处理（mashup），如果用另一个近来流行的词来说的话。每个旗帜广告（banner ad)都是用来在两个网站之前无缝合作，向位于另外一台计算机上的读者传递一个整合好的页面。   
  
　　Akamai也将网络看作平台，并且在一个更深入的层次上，来搭建一个透明的缓存和内容分发网络，以便降低宽带的拥塞程度。   
  
　　虽然如此，这些先驱提供了有益的对比，因为后来者遇到同样问题的时候，可以将先驱们的解决方案进一步延伸，从而对新平台本质的理解也更为深刻了。DoubleClick和Akamai都是Web 2.0的先驱，同时我们也可以看到，可以通过引入更多Web 2.0的设计模式，来实现更多的应用。   
  
　　让我们对这三个案例中的每一个都作一番深究，来探讨其间的一些本质性的差别。   
  
　　Netscape 对 Google   
  
　　如果Netscape可以称为Web 1.0的旗手，那么Google几乎可以肯定是Web 2.0的旗手，只要看看他们的首次公开上市（IPO)是如何地揭示了各自的时代就清楚了。所以我们就从这两个公司和其定位的差别入手。   
  
　　Netscape以传统的软件摹本来勾勒其所谓“互联网作为平台”：他们的旗舰产品是互联网浏览器，一个桌面应用程序。同时，他们的战略是利用他们在浏览器市场的统治地位，来为其昂贵的服务器产品建立起市场。从理论上讲，在浏览器中控制显示内容和程序的标准，赋予了Netscape一种市场支配力，如同微软公司在个人计算机市场上所享受的一样。很像当初“自行的马车”（horseless carriage）将汽车描绘为一种熟知事物的延伸，Netscape曾推销一种网络桌面（webtop）来替代传统的桌面（desktop），并且计划借助信息更新，以及由购买了Netscape服务器的信息提供者来推送的各种小程序，来开发推广这种网络桌面。   
  
　　最终，浏览器和网络服务器都变成了“日用品”，同时价值链条也向上移动到了在互联网平台上传递的服务。   
  
　　作为对比，Google则以天生的网络应用程序的角色问世，它从不出售或者打包其程序，而是以服务的方式来传递。客户们直接或间接地为其所使用的服务向Google付费。原有软件工业缺陷荡然无存。没有了定期的软件发布，只需要持续的改善。没有了许可证或销售，只需要使用。没有了为了让用户在其设备上运行软件而不得不进行的平台迁移，只需要搭建宏大的、由众多个人计算机组成的、可伸缩的网络，其上运行开源操作系统，及其及自行研制的应用程序和工具，而公司之外的任何人则永远无法接触到这些东西。   
  
　　在其底层，Google需要一种Netscape从未需要过的能力：数据库管理。Google远远不只是一个软件工具的集合，它是一个专业化的数据库。没有这些数据，那些工具将毫无用武之地；没有这些软件，数据也将无可控制。软件许可证制度和对应用程序接口（API）的控制——上一个时代的法宝——已经毫不相关了，因为Google的软件只需要执行而从不需要分发，也因为如果不具备收集和管理数据的能力，软件本身就没有什么用处了。事实上，软件的价值是同它所协助管理的数据的规模和活性成正比的。   
  
　　Google的服务不是一个简单的服务器，虽然其服务是通过大规模的互联网服务器集合来传递的；其服务也不是一个浏览器，虽然这种服务是被用户在浏览器中体验到的。Google的旗舰产品——搜索服务，甚至不托管它让用户来搜寻的内容。很像一个电话通话过程，不仅发生在通话的两端，而且发生在中间的网络上。作为用户和其在线体验的一个中介，Google作用于浏览器、搜索引擎和最终的内容服务器之间的空间中。   
  
　　虽然Netscape和Google都可以被描述为软件公司，但显然Netscape可以归到Lotus，Microsoft，Oracle，[SAP](http://www.enet.com.cn/cio/inforcenter/S4654_20050420410088.html" \t "_blank)，以及其他发源于上个世纪八十年代软件革命的那些公司所组成的软件世界。而Google的同伴们，则是像eBay，Amazon，Napster，及至DoubleClick和Akamai这样的互联网公司。   
  
　　DoubleClick对Overture和AdSense   
  
　　同Google类似，DoubleClick是一个名副其实的互联网时代的孩子。它把软件作为一种服务，在数据管理方面具有核心竞争力，并且正如上文所述，它是一个早在连网络服务的名字还不曾有的时候，就已然开始其服务的先驱。然而，DoubleClick最终还是被其商业模式局限住了。它所贯彻的是九十年代的互联网观念。这种观念围绕着出版，而不是参与；围绕着广告客户，而不是消费者，来进行操纵；围绕着规模，认为互联网会被如MediaMetrix等网络广告评测公司尺度下的所谓顶级网站所统治。   
  
　　结果是，DoubleClick得意地在其网站上引用道：“超过2000种的成功应用”。而相对比的是，Yahoo!公司的搜索市场（从前的Overture）和Google的AdSense产品，已经在为几十万的广告客户服务。   
  
　　Overture和Google的成功源自于对克里斯·安德森（Chris Anderson）提到的所谓“长尾”的领悟，即众多小网站集体的力量提供了互联网的大多数内容。DoubleClick的产品要求一种签订正式的销售合同，并将其市场局限于很少的几千个大型网站。Overture和Google则领会到如何将广告放置到几乎所有网页上。更进一步地，它们回避了发行商和广告代理们所喜爱的广告形式，例如旗帜广告和弹出式广告，而采用了干扰最小的、上下文敏感的、对用户友好的文字广告形式。   
  
　　Web 2.0的经验是：有效利用消费者的自助服务和算法上的数据管理，以便能够将触角延伸至整个互联网，延伸至各个边缘而不仅仅是中心，延伸至长尾而不仅仅是头部。   
  
　　毫不奇怪，其他Web 2.0的成功故事也显示着同样的轨迹。eBay扮演着一个自动的中间媒介的角色，使个体之间发生的几个美元的偶然性的交易成为可能。Napster（虽然已经出于法律原因而关闭）将其网络建立在一个集中的歌曲数据库之上，但是它让每一个下载者都成为一台服务器，从而使其网络逐渐扩大。   
  
　　Akamai 对 BitTorrent   
  
　　同DoubleClick类似，Akamai的业务重点面向网络的头部，而不是尾部；面向中心，而不是边缘。虽然它服务于那些处于网络边缘的个体的利益，为他们访问位于互联网中心的高需求的网站铺平了道路，但它的收入仍然来自从那些位于中心的网站。   
  
　　BitTorrent，像P2P风潮中的其他倡导者一样，采用了一种激进的方式来达到互联网去中心化（internet decentralization）的目的。每个客户端同时也是一个服务器；文件被分割成许多片段，从而可以由网络上的多个地方提供，透明地利用了网络的下载者来为其他下载者提供带宽和数据。事实上，文件越流行下载得越快，因为有更多的用户在为这个文件提供带宽和各个片段。   
  
　　BitTorrent由此显示出Web 2.0的一个关键原则：用户越多，服务越好。一边是Akamai必须增加服务器来改善服务，另一边是BitTorrent用户将各自的资源贡献给大家。可以说，有一种隐性的“参与体系”内置在合作准则中。在这种参与体系中，服务主要扮演着一个智能代理的作用，将网络上的各个边缘连接起来，同时充分利用了用户自身的力量。   
  
　　2. 利用集体智慧   
  
　　在诞生于Web 1.0时代并且存活了下来，而且要继续领导Web 2.0时代的那些巨人的成功故事的背后，有一个核心原则，就是他们借助了网络的力量来利用集体智慧：   
  
　　--超级链接是互联网的基础。当用户添加新的内容和新的网站的时候，将被限定在一种特定的网络结构中，这种网络结构是由其他用户发现内容并建立链接的。如同大脑中的神经突触，随着彼此的联系通过复制和强化变得越来越强，而作为所有网络用户的所有活动的直接结果，互联的网络将有机地成长。   
  
　　--Yahoo!是第首例伟大的成功故事，诞生于一个分类目录，或者说是链接目录，一个对数万甚至数百[万网](http://www.enet.com.cn/cio/inforcenter/S5131_20050722437515.html)络用户的最精彩作品的汇总。虽然后来Yahoo!进入了创建五花八门的内容的业务，但其作为一个门户来收集网络用户们集体作品的角色，依然是其价值核心。   
  
　　--Google在搜索方面的突破在于PageRank技术，该技术令其迅速成为搜索市场上毫无争议的领导者。PageRank是一种利用了网络的链接结构，而不是仅仅是使用文档的属性，来实现更好的搜索效果的方法。   
  
　　--eBay的产品是其全部用户的集体活动，就向网络自身一样，eBay随着用户的活动而有机地成长，而且该公司的角色是作为一个特定环境的促成者，而用户的行动就发生在这种环境之中。更重要的是，eBay的竞争优势几乎都来自于关键性的大量的买家和卖家双方，而这正是这一点使得后面许多竞争者的产品的吸引力显著减低。   
  
　　--Amazon销售同Barnesandnoble.com等竞争者相同的产品，同时这些公司从卖方获得的是同样的产品描述、封面[图片](http://www.enet.com.cn/pic/)和目录。所不同的是，Amazon已然缔造出了一门关于激发用户参与的科学。Amazon拥有比其竞争者高出一个数量级以上的用户评价，以及更多的邀请来让用户以五花八门的方式，在近乎所有的页面上进行参与，而更为重要的是，他们利用用户的活动来产生更好的搜索结果。Barnesandnoble.com的搜索结果很可能指向该公司自己的产品，或者是赞助商的结果，而Amazon则始终以所谓“最流行的”打头，这是一种实时计算，不仅基于销售，而且基于其他一些被Amazon内部人士称为围绕着产品“流动”（flow）的因素。由于拥有高出对手一个数量级的用户参与，Amazon销售额超出竞争对手也就不足为奇了。   
  
　　现在，具备了这种洞察力，并且可能会将之延伸开来的那些创新型的公司，正在互联网上留下他们的印迹。   
  
维基百科全书（Wikipedia）是一种在线百科全书，其实现基于一种看似不可能的观念。该观念认为一个条目可以被任何互联网用户所添加，同时可以被其他任何人编辑。无疑，这是对信任的一种极端的实验，将埃里克·雷蒙德（Eric Raymond）的格言（源自开放源码软件的背景之下）：“有足够的眼球，所有的程序缺陷都是肤浅的”（with enough eyeballs, all bugs are shallow）运用到了内容的创建之中。维基百科全书已然高居世界网站百强之列，并且许多人认为它不久就将位列十强。这在内容创建方面是一种深远的变革。   
  
　　像del.icio.us（美味书签）和Flickr这样的网站，其公司已经在近期获得了广泛的关注，并且已经在一种被人们成为“分众分类”（folksonomy，有别于传统分类法）的概念上成为先行者。“分众分类”是一种使用用户自由选择的关键词对网站进行协作分类的方式，而这些关键词一般称为标签（tags)。标签化运用了像大脑本身所使用的那种多重的、重叠的关联，而不是死板的分类。举一个经典的例子，在Flickr网站上，一幅小狗照片可能被加上“小狗”和“可爱”这样的标签，从而允许系统依照用户行为所产生的自然的方式来进行检索。   
  
　　协作式垃圾信息过滤产品，例如Cloudmark，就聚集了电子邮件用户们对于“一封邮件是或者不是垃圾邮件”的众多相互独立的决策，从而胜过了依赖于分析邮件本身的那些系统。   
  
　　伟大的互联网成功者并不主动地到处推销其产品，这几乎成为公理。他们采用“病毒式营销”（viral marketing）的方式，也就是说，一些推介会直接从一个用户传播到另外一个用户。如何一个网站或产品依赖广告来进行宣传，你几乎可以断定它不是Web 2.0。   
  
　　即便许多互联网基础设施本身，包括在大多数网络服务器中用到的Linux，Apache，MySQL，以及Perl，PHP或Python代码，也都依靠开放源码的对等生产（peer-production）的方式。其中包含了一种集体的、网络赋予的智慧。在SourceForge.net网站上列有至少10万种开放源码软件项目。任何人都可以添加一个项目，任何人都可以下载并使用项目代码。   
  
　　同时，由于作为用户使用的结果，新的项目从边缘迁移到中心。一个对软件的有机的接受过程几乎完全依靠病毒式营销。同时，作为用户应用的结果，新的项目从边缘迁移到中心，这是一种几乎完全依靠病毒式营销的，有机的软件采用过程，。   
  
　　经验是：源于用户贡献的网络效应，是在Web 2.0时代中统治市场的关键。

博客和大众智慧   
  
　　Web 2.0时代一项最受追捧的特性就是博客的兴起。个人主页从互联网早期就已经存在了，而个人日记和每日发表观点的专栏就更渊源久远了，那么到底有什么让人大惊小怪的呢？   
  
　　归根底地，博客只是一种日记形式个人网页。但正如里奇·斯格仁塔（Rich Skrenta）指出的，博客的按时间顺序来排列的结构“看起来像是一个微不足道的变化，但却推动着一个迥然不同的分发、广告和价值链。”   
  
　　其中一大变化就是一项称为RSS的技术。RSS是自早期计算机高手们认识到CGI（公共网关接口）可用来创建以数据库为基础的网站以来，在互联网根本结构方面最重要的进步。RSS使人们不仅仅链接到一个网页，而且可以订阅这个网页，从而每当该页面产生了变化时都会得到通知。斯格仁塔将之称为“增量的互联网”（incremental web）。其他人则称之为“鲜活的互联网”（live web）。   
  
　　当然，现在所谓“动态网站”（即具有动态产生的内容的、由数据库驱动的网站）取代了十年前的静态网站。而动态网站的活力不仅在于网页，而且在链接方面。一个指向网络博客的链接实际上是指向一个不断更新的网页，包括指向其中任何一篇文章的“固定链接”（permalinks），以及每一次更新的通知。因此，一个RSS是比书签或者指向一个单独网页的链接要强大得多。   
  
　　RSS同时也意味着网页浏览器不再只是限于浏览网页的工具。尽管诸如Bloglines之类的RSS聚合器（RSS aggregators）是基于网络的，但其他的则是桌面程序，此外还有一些则可以用在便携设备上来接受定期更新的内容。   
  
　　RSS现在不仅用于推送新的博客文章的通知，还可以用于其他各种各样的数据更新，包括股票[报价](http://price.enet.com.cn/)、天气情况、以及图片。这类应用实际上是对RSS本源的一种回归：RSS诞生于1997年，是如下两种技术的汇合：一种是戴夫·温纳（Dave Winer）的“真正简单的聚合”（Really Simple Syndication）技术，用于通知博客的更新情况；另一种是Netscape公司提供的“丰富站点摘要”（Rich Site Summary）技术，该技术允许用户用定期更新的数据流来定制Netscape主页。后来Netscape公司失去了兴趣，这种技术便由温纳的一个博客先驱公司Userland承接下来。不过，在现在的应用程序实现中，我可以看出两者共同的作用。   
  
　　但是，RSS只是令博客区别于同普通网页的一部分原因。汤姆·科特斯（Tom Coates）这样评论固定链接的重要性：   
  
　　“现在它可能看上去像是一项普普通通的功能，但它却有效地将博客从一个易于发布（ease-of-publishing）的现象，进一步转变为互相交叉的[社区](http://bbs.enet.com.cn/)的一种对话式的参与。这是首次使得对其他人的网站上的很特定的帖子表态和谈论变得如此地容易。讨论出现了，聊天也出现。同时，其结果是出现了友谊或者友谊更加坚定了。固定链接是第一次也是最为成功的一次在博客之间搭建桥梁的尝试。”   
  
　　在许多方面，RSS同固定链接的结合，为HTPP（互联网协议）增添了NNTP（新闻组的网络新闻协议）的许多特性。所谓“博客圈”（blogosphere），可以将其视作一种同互联网早期的、以对话方式来灌水的新闻组和公告牌相比来说，新型的对等（peer-to-peer）意义上的等价现象。人们不仅可以相互订阅网站并方便地链接到一个页面上的特定评论，而且通过一种称为引用通告（trackbacks）的机制，可以得知其他任何人链接到了他们的页面，并且可以用相互链接或者添加评论的方式来做出回应。   
  
　　有趣的是，这种双向链接（two-way links）曾是象Xanadu之类的早期超文本系统的目标。超文本纯粹论者已然将引用通告颂扬为向双向链接迈进了一步。但需要注意的是，引用通告不是一个真正的双向链接，确切地讲是一种（潜在地）实现了双向链接效果的对称式单向链接。其间的区别看起来可能很细微，但实际上却是巨大的。诸如Friendster, Orkut和LinkedIn那样的社交网络系统（social networking systems），需要接受方做出确认以便建立某种连接，从而缺少像互联网架构本身那样的可伸缩性。正如照片共享服务Flickr网站的创始人之一卡特里纳·费克（Caterina Fake）所指出的，注意力仅在碰巧时才礼尚往来。（Flickr因此允许用户设置观察列表，即任何用户都可以通过RSS来订阅其他所有用户的照片流。注意的对象将会被通知，但并不一定要认可这种连接。）  
  
　　如果Web 2.0的一个本质是利用集体智慧，来将互联网调试为一种所谓的全球的大脑，那么博客圈就是前脑中喋喋不休的呓语，那种我们整个头脑中都能听到的声音。这可能并不反映出大脑的往往是无意识的深层结构，但却是一种有意识的思考的等价物。作为一种有意识的思考和注意力的反映，博客圈已经开始具有强有力的影响。   
  
　　首先，因为搜索引擎使用链接结构来辅助预测有用的页面，作为最多产和最及时的链接者，博客们在修整搜索引擎结果方面充当着一种不成比例的角色。其次，因为博客社区是如此多地自相引用，关注其他博客的博客们开阔了他们的视野和能力。此外，评论家们所批判的“回音室”（echo chamber）也是一种放大器。   
  
　　如果只是一种放大器，那么撰写博客将会变得无趣。但是像维基百科全书一样，博客将集体智慧用作一种过滤器。被詹姆士·苏瑞奥维奇（James Suriowecki）称为“大众智慧”（the wisdom of crowds）的规律起了作用，并且就像PageRank技术所产生的结果胜过分析任何单一文档一样，博客圈的集体关注会筛选出有价值的东西。   
  
　　虽然主流媒体可能将个别的博客视为竞争者，但真正使其紧张的将是同作为一个整体的博客圈的竞争。这不仅是网站之间的竞争，而且是一种商业模式之间的竞争。Web 2.0的世界也正是丹·吉尔默（Dan Gillmor）的所谓“个人媒体”（We，the media)的世界。在这个世界中，是所谓“原本的听众”，而不是密实里的少数几个人，来决定着什么是重要的。   
  
　　3. 数据是下一个Intel Inside   
  
　　现在每一个重要的互联网应用程序都由一个专门的数据库驱动：Google的网络爬虫, Yahoo!的目录（和网络爬虫），Amazon的产品数据库，eBay的产品数据库和销售商，MapQuest的地图数据库，Napster的分布式歌曲库。正如哈尔·瓦里安（Hal Varian）在去年的私人对话中谈到的，“SQL是新的HTML”。数据库管理是Web 2.0公司的核心竞争力，其重要性使得我们有时候称这些程序为“讯件”（infoware）而不仅仅是软件。   
  
　　该事实也引出了一个关键问题：谁拥有数据？   
  
　　在互联网时代，我们可能已经见到了这样一些案例，其中对数据库的掌控导致了对市场的支配和巨大的经济回报。当初由美国政府的法令授权给Network Solutions公司（后被Verisign公司收购）的对域名注册的垄断，曾经是互联网上的第一个摇钱树。虽然我们在争论通过控制软件的API来形成商业优势在互联网时代会变得困难得多，但是对关键数据资源的控制则不同，特别是当要创建这些数据资源非常昂贵，或者经由网络效应容易增加回报的时候。   
  
　　注意一下由MapQuest, maps.yahoo.com，maps.msn.com，或者maps.google.com等网站提供的每张地图下面的版权声明，你会发现这样一行字“地图版权NavTeq，TeleAtlas”，或者如果使用的是新的卫星图像服务，则会看到“图像版权Digital Globe”的字样。这些公司对其数据库进行了大量的投资。（仅NavTeq一家，就公布投资7.5亿美元用于创建其街道地址和路线数据库。Digital Globe则投资5亿美元来启动其自有卫星，来对政府提供的图像进行改进。）NavTeq竟然已做了很多模仿Intel的耳熟能详的Intel Inside标识的事：例如带有导航系统的汽车就带有“NavTeq Onboard”的印记。数据是许多此类程序事实上的Intel Inside，是一些系统的唯一的信息源组件，这些系统的软件体系多数是开放源码的，也有商业化的。   
  
　　当前竞争火热的网络地图（web mapping）领域显示着，对拥有软件核心数据的重要性的疏忽大意，将最终削弱其竞争地位。MapQuest在1995年率先进入地图领域，随后是Yahoo!，再后来是Microsoft，而最近Google也决定挺进这一市场，他们可以轻松地通过对同一数据的授权来提供一个具有竞争力的程序。   
  
　　然而，作为对比的是Amazon.com的竞争地位。像Barnesandnoble.com这样的竞争者一样，其原始数据库来自于ISBN注册商.R. Bowker。但是同MapQuest不同，Amazon大力增强其数据，增加出版商提供的数据，例如封面图片，目录，索引，和样张材料。更重要的是，他们利用了其用户来评注数据，以至于十年之后，是Amazon而不是Bowker，成为图书文献信息的主要来源，一个学者、图书管理员和消费者的参考书目来源。Amazon还引入了其专有的标识符，即ASIN，该标识符在ISBN存在时与之对应，而当产品不带有ISBN时，就创建出一个等价的命名空间。Amazon从而有效地“吸收和拓展了”其数据提供商。   
  
　　设想如果MapQuest也已做了同样的事情，利用他们的用户来评注地图和路线，添加新的价值层面。那么对仅仅通过授权使用基础数据来进入这一市场的其他竞争者，将造成远远大得多的困难。   
  
　　近期Google地图的引入，为应用程序销售商和其数据提供商之间的竞争，提供了一个活生生的实验室。Google的轻量型编程模型已经引发了不计其数的增值服务的出现，这些服务以数据混合的方式，将Google的地图同其他可以通过互联网访问的数据源相结合。保罗·拉特马赫（Paul Rademacher）的housingmaps.com是这种混合的一个上佳范例，其网站将Google的地图同Craigslist的公寓出租，以及住宅购买数据相结合，来创建一种交互式的房屋搜索工具。   
  
　　目前，这些混合大多是由程序高手们实现的创新性的实验产品。但是企业行动将紧随其后。并且，人们已经可以从至少一类开发者中发现这一点。Google已经将数据源提供者的角色从Navteq那里夺走，并且将自己定位为一个令人喜爱的中介者。在以后几年里，我们将会看到数据提供商和程序销售商之间的斗争，因为两大阵营都认识到了，特定的数据类别在作为搭建Web 2.0程序的积木时是多么的重要。   
  
　　这场竞赛已经涉及到拥有特定类别的核心数据：位置、身份、公共事件日历、产品标识和命名空间等。在许多情况下，在那些创建数据需要巨额成本的地方，也可能存在一种如同Intel Inside方式一样凭借单一数据源来所有作为的机遇。其他情况下，胜者将是那些通过用户聚合来达到临界规模，并且将聚合的数据融入系统服务中的公司。   
  
　　比如，在身份标识领域，PayPal，Amazon的一键式，以及拥有数百万用户的交流系统，都有可能成为创建整个网络范围的身份标识数据库的正当竞争者。（关于此，Google最近使用手机号码作为Gmail账号标识的尝试，可能就是朝借鉴和拓展电话系统所迈出的一步。）同时，像Sxip这样的创业公司，正在探索联合身份标识的可能性，以寻求一种“分布一键式”，从而提供一个无缝的Web 2.0标识子系统。在日历领域，EVDB则是通过维基式参与体系来搭建世界上最大的共享日历的一种尝试。虽然评判者尚在观望着任何一个特定创业公司或方式的成功是否，但很显然，这些领域的标准和解决方案，有效地将某些数据转变为“互联网操作系统”（internet operating system）的可靠的子系统，并将促成下一代的应用程序。   
  
　　关于数据，必须注意一个进一步的方面，那就是用户关心其隐私和对自己的数据的权限。在许多早期的网络程序中，版权只被松散地执行。例如，Amazon宣称对任何提交到其网站的评论的所有权，但却缺少强制性，人们可以将同样的评论转贴到其他任何地方。然而，随着很多公司开始认识到，对数据的掌控有可能成为他们首要的竞争优势来源，我们将会看到在此类控制方面强度更大的尝试。   
  
　　正如专有软件的增长而导致自由软件运动一样，在下一个10年中我们会看到专有数据库的增长将导致自由数据运动。在像维基百科全书这样的开放数据项目、创作共用（Creative Commons）、以及像Greasemonkey（让用户决定如何在其计算机上显示数据）这样的软件项目中，我们可以看到这种对抗势头的前兆。

4. 软件发布周期的终结   
  
　　如上文在对Google和Netscape的比较中谈到的，互联网时代软件的代表性特征就是它应该被作为服务来交付。这种事实导致这类公司的商业模式上很多根本性的变化。   
  
　　1. 运营必须成为一种核心竞争力。Google或者Yahoo!在产品开发方面的专门技术，必须同日常运营方面的专门技术相匹配。从软件作为制造品到软件作为服务的变化是如此地根本，以至于软件将不再能完成任务，除非每日加以维护。Google必须持续抓取互联网并更新其索引，持续滤掉链接垃圾和其他影响其结果的东西，持续并且动态地响应数千万异步的用户查询，并同步地将这些查询同上下文相关的广告相匹配。   
  
　　所以，Google的系统管理、网络、和负载均衡技术，可能比其搜索算法更被严加看管，也就不足为奇了。Google在自动化这些步骤上的成功是其同竞争者相比更有成本优势的一个关键方面。   
  
　　同样也不足为奇的是，像Perl、Python、PHP、和当前的Ruby这样的脚本语言在Web 2.0公司中扮演着重要角色。Sun公司的第一个网管哈桑·施罗德（Hassan Schroeder）曾对Perl有一个著名的形容：“互联网的管道胶带”（the duct tape of the internet）。事实上，动态语言（常常被称为脚本语言，并被软件制品时代的软件工程师所贬低），是系统和网络管理员，以及创建可经常更新的动态系统的程序开发者们所喜爱的工具。   
  
　　2. 用户必须被作为共同开发者来对待，这是从对开放源码开发实践的一种反思中得出的（即便所涉及的软件不太可能以开放源码授权方式来发行）。开放源码的格言“早发布并常发布”（release early and release often）事实上已经演变成一种更为极端的定位“永远的测试版”（the perpetual beta）。其中产品在开放状态下开发，新的功能以每月、每周、甚至每天的速度被加入进来。Gmail、Google Maps、Flickr、del.icio.us，和其他类似的服务，可能会在某个阶段打着测试版的标识多年。   
  
　　故此，实时地监测用户行为，来考察哪些新特性被使用了，以及如何被使用的，将成为另外一种必须的核心竞争力。一位工作于一个主要在线服务网络商的开发者评论道：“我们每天在网站的某些部分提供两到三个新的特性，而且如果用户不采用它们，我们就将其撤掉。如果用户喜欢它们，我们就将其推广到整个网站。”   
  
　　Flickr的总开发师卡尔·亨德森（Cal Henderson），近来透露了他们是如何在短至每半个小时就部署一个新版本的。显而易见，这是同传统方式有天壤之别的开发模式。虽然不是所有的网络程序都以像Flickr这样的极端方式来开发，但几乎所有网络程序都有一个同任何PC或者客户-服务器时代截然不同的开发周期。正因如此，ZDnet杂志才论断Microsoft不会打败Google：“Microsoft的商业模式依赖于每个人在每两到三年都升级他们的计算环境。Google的模式则依靠任何人每天在其计算环境中自行探索新东西。”   
  
　　虽然Microsoft已经体现了从竞争中学习并最终做得最好的强大能力，但是毫无疑问这一次的竞争要求Microsoft（可以扩展到任何现存的软件公司）来成为一种在深入层面上显著有别的公司。天生的Web 2.0公司在享受自然而然的优势，因为它们不需要去摆脱陈旧的模式（及其相应的商业模式和营收来源）。   
  
　　5. 轻量型编程模型   
  
　　一旦网络服务的观念深入人心，大型公司将以复杂的网络服务堆栈来加入到纷争之中。这种网络服务堆栈被设计用来为分布式程序建立更可靠性的编程环境。   
  
　　但是，就像互联网成功正是因为它推翻了许多超文本理论一样，RSS以完美的设计来取代简单的实用主义，已经因其简单性而成为大概是应用最广泛的网络服务，而那些复杂的企业网络服务尚未能实现广泛的应用。   
  
　　类似地，Amazon.com的网络服务有两种形式：一种坚持SOAP(Simple Object Access Protocol，简单对象访问协议）网络服务堆栈的形式主义；另一种则简单地在HTTP协议之外提供XML数据，这在轻量型方式中有时被称为REST（Representational State Transfer，代表性状态传输）。虽然商业价值更高的B2B连接（例如那些在Amazon和一些像ToysRUs这样的零售伙伴之间的连接）使用SOAP堆栈，但是根据Amazon的报道，95%的使用来自于轻量型REST服务。   
  
　　同样的对简易性的要求，可以从其他“朴实的”网络服务中见到。Google近来的Google地图的推出就是一个例子。Google地图的简单AJAX（Javascript和XML的结合）接口迅速被程序高手们破译，被随即进一步将其数据混合到新的服务之中。   
  
　　地图相关网络服务已经存在了一段时间，例如像ESRI那样的GIS（地理信息系统），以及从MapQuest和Microsoft的MapPoint。但是Google地图以其简洁性而让世界兴奋起来。虽然从前销售商所支持的网络服务都要求各方之间的正式约定，但Google地图的实现方式使数据可以被捕获，于是程序高手们很快就发现了创造性地重用这些数据的方法。   
  
　　这里有几条重要的经验：   
  
　　1. 支持允许松散结合系统的轻量型的编程模型。由企业开发的网络服务堆栈的复杂设计是用来促成紧密结合的。虽然这在许多情况下是必须是，但是许多最重要的应用程序可以事实上保持松散结合，甚至是脆弱的结合。Web 2.0的理念同传统的IT的理念迥然不同。   
  
　　2. 考虑聚合（syndication）而不是协调（coordination）。简单的网络服务，例如RSS和基于REST的网络服务，是用来向外聚合数据，但并不控制其达到连接的另外一端时发生的事情。这种想法是互联网本身的基础，一种对所谓端到端原则的反映。   
  
　　3. 可编程性和可混合性设计。像最初的互联网一样，RSS和AJAX这样的系统，都有此共同点：重用的障碍非常低。许多有用的软件事实上是开放源码的，而即便它不是，也没有许多东西来保护其知识产权。互联网浏览器的“查看源文件”选项，使得许多用户可以复制其他任何用户的网页；RSS被设计得使用户能够在需要的时候查看所需要的内容，而不是按照信息提供者的要求；最成功的网络服务，是那些最容易采纳未被服务创建者想到的新的方向。同更普遍的“保留所有权利”（all rights reserved）相比，随着创作共用约定而普及的“保留部分权利”（Some Rights Reserved）一词成为一个有益的指路牌。   
  
　　装配中的创新   
  
　　轻量型商业模型是对轻量型编程和轻量型结合的一种自然产物。Web 2.0的理念善于重用。一种像housingmaps.com这样的新服务，是通过将两个现存服务抓取到一起来简单地创建起来的。Housingmaps.com还没有商业模式（目前为止），但对于许多小规模的服务，Google的AdSense（或Amazon的associates fees计划，或者两者都是）为同类服务提供了营收模式。   
  
　　这些案例为Web 2.0的另外一个关键原则提供了启发，我们将之称为“装配中的创新”。当商品组件充裕时，你可以通过以新颖的或者有效的方式来装配这些组件来创建价值。很像PC革命为硬件商品装配提供了许多创新的机会，其中像Dell这样的公司创造了这种装配的科学，并从而打败了那些商业模式上要求产品开发方面的创新的公司，我们相信Web 2.0为各个公司提供了，通过在利用和整合由其他人提供的服务方面逐渐完善，来赢得竞争的机会。   
  
　　6. 软件超越单一设备   
  
　　另外一个值得一提的Web 2.0特性是Web 2.0已经不再局限于PC平台这样一个事实。在对Microsoft的告别建议中，长期的Microsoft开发者戴夫·斯塔兹（Dave Stutz）指出：“超越单一设备而编写的有用软件将在未来很长一段时间里获得更高的利润”。   
  
　　当然，任何的网络程序都可被视为超越单一设备的软件。毕竟，即便是最简单的互联网程序也涉及至少两台计算机：一个负责网络服务器，而另一个负责浏览器。而且就如我们已经探讨过的，在将网络作为平台的开发中，把这个概念拓展到由多台计算机提供的服务而组成的合成应用程序中。   
  
　　但是如同Web 2.0的许多领域一样，在那些领域中“2.0版的事物”（2.0-ness）并不是全新的，而是对互联网平台真正潜能的一种更完美的实现，软件超越单一设备这一说法赋予我们为新平台设计程序和服务的关键性的洞察力。   
  
　　迄今为止，iTunes是这一原则的最佳范例。该程序无缝地从掌上设备延伸到巨大的互联网后台，其中PC扮演着一个本地缓存和控制站点的角色。之前已经有许多将互联网的内容带到便携设备的尝试，但是iPod/iTunes组合却是这类应用中第一个从开始就被设计用于跨越多种设备的。TiVo则是另外一个不错的例子。   
  
　　iTunes和TiVo也体现了Web 2.0的其他一些核心原则。它们本身都不是网络程序，但都利用了互联网平台的力量，使网络成为其体系中无缝连接的、几乎不可察觉的一部分。数据管理显然是它们所提供的价值的核心。它们也是服务，而非打包的程序（虽然对于iTunes来说，它可以被用作一个打包的程序来仅仅管理用户本地的数据）。不仅如此，TiVo和iTunes都展示了一些集体智慧的方兴未艾的应用。虽然对于每个情况，其实验都是同网络IP入口的周旋。iTunes中只有有限的参与体系，虽然近来增加的播客（podcasting）将这一规则规律性了不少。   
  
　　这正是我们希望看到伟大变革的Web 2.0领域中的一个，随着越来越多的设备正连接到这个新的平台中来。当我们的电话和汽车虽不消费数据但却报告数据时，可能会出现什么样的程序呢？实时的交通监测、快闪暴走族（flash mobs）、以及公民媒体，只不过是新平台的能力的几个早期警示。

　7. 丰富的用户体验   
  
　　最早可以追溯到1992年魏裴（Pei Wei）开发的Viola浏览器，互联网就被用来在网页浏览器中传送“小程序”（applet）和其他一些活动内容。1995年Java的引入就是围绕着这样的小程序的传送。JavaScript和后来的DHTML都被作为轻量型方式引入，来为客户端提供可编程性和丰富的用户体验。几年以前，Macromedia缔造出“丰富的互联网应用程序”（Rich Internet Applications）一词（该词也被Flash的竞争者开放源码的Laszlo系统使用），以便凸显Flash不仅可传送多媒体内容，而且可以是GUI（图形用户界面）方式的应用程序体验。   
  
　　然而，互联网传递整个应用程序的潜能在Google引入Gmail之前，一直没有成为主流，紧接着就是Google地图程序，一些基于互联网的带有丰富用户界面以及PC程序等同的交互性的应用程序。在网络设计公司Adaptive Path的耶希&#8226;詹姆斯&#8226;加莱特（Jesse James Garrett）的一个讨论会论文中，Google所使用的这组技术被命名为AJAX。他写道：   
  
　　Ajax不是一项技术。它其实是几项技术，每项技术自身都很繁荣，它们以强有力的全新方式结合起来。Ajax涵盖：   
-- 运用XHTML和CSS实现基于各种标准的展示。   
-- 运用文档对象模型（Document Object Model）实现动态显示和交互。   
-- 运用XML和XSLT实现数据交换和操作。   
-- 运用XMLHttpRequest实现异步数据检索。   
-- JavaScript将所有这些绑定到一起。   
  
　　AJAX也是Web 2.0程序的一个关键组件，例如现在归属Yahoo!的Flickr，37signals的程序basecamp和backpack，以及其他Google程序，例如Gmail和Orkut。我们正在步入一个史无前例的用户界面创新阶段，因为互联网开发者们终于可以创建，像本地基于PC的应用程序一样丰富的网络程序了。   
  
　　有趣的是，许多现在正被探索的功能已经存在了很多年了。90年代后期，Microsoft和Netscape，都对现在终于被认识到的那些功能有所洞察，但是它们对于所要采用的标准的争斗，使得实现跨浏览器的应用程序变得困难。仅在当初Microsoft确定无疑地赢得了浏览器之战的时候，而且那时事实上只需要针对一个浏览器标准，编写这种程序才成为可能。同时，虽然Firefox在浏览器市场中重新引入了竞争，但至少在目前我们还没有看到对互联网标准的破坏性的争夺以至于我们倒退到90年代。   
  
　　在接下来的几年中，我们会看到许多新的网络程序，不仅确实是新颖的程序，而且是对PC程序丰富的网络再现。到目前为止，每个平台的变革也都为改变那些在旧平台中占主导地位的程序的领导地位创造了机会。   
  
　　Gmail已经在电子邮件中提供了一些有意思的创新，将互联网的力量（随处可访问、深层的数据库能力、可搜索性）与在易用性方面同PC界面接近的用户界面相结合。同时， PC平台上的其他邮件程序，正在从另一端通过增添IM和呈现能力，来蚕食着这一领域。我们离集成通信客户端有多少远呢？这些集成通信客户端应是整合了电子邮件、即时通信和手机，并且应使用VoIP以便向网络程序的丰富功能中添加语音能力。这种竞赛已经开始。   
  
　　我们也很容易看到Web 2.0是如何重新打造地址簿的。一个Web 2.0风格的地址薄将把PC或电话上的本地地址簿，仅仅当作一种你显式要求系统记忆的联系人的缓存。同时，一个基于互联网的Gmail风格的异步代理，将保存发送或者接收的每个消息，每个电子邮件地址和每个使用过的电话号码，并且创造出社交网络的启发性算法，来决定当一个答案不能在本地缓存中找到时，应该提供哪个作为替代。在缺少答案的情况下，该系统会查询更广阔的社交网络。   
  
　　一个Web 2.0的字处理程序将会支持维基风格的协作编辑，而不仅仅是处理独立的文档。但是该程序也会支持我们期望在基于PC的字处理器中得到的那种丰富格式。Writely是这种程序的一个优秀范例，虽然它尚未引起广泛关注。   
  
　　此外，Web 2.0革命不会局限于PC程序。例如，在CRM这样的企业级应用程序中，Salesforce.com展示了网络是如何被用来以服务的方式来传递软件的。   
  
　　对新的进入者来说，竞争机会在于充分开发Web 2.0的潜能。成功的公司将创建可以向其用户学习的程序，利用可供参与的体系来建立一种决定性的优势，不仅在软件的界面方面，而且在共享数据的丰富程度方面。   
  
  
　　Web 2.0公司的核心竞争力   
　　   
　　在探索上述七大原则的过程中，我们已经强调了Web 2.0的一些主要特性。我们探讨的每一个例子都体现着这些原则中的一个或多个，但是可能不满足其他的原则。因此，让我们通过总结我们认为是Web 2.0公司核心竞争力的一些方面，来结束本文。   
  
-- 服务，而不是打包的软件，具有高成本效益的可伸缩性。   
-- 控制独特的、难以再造的数据源，并且用户越多内容越丰富。   
-- 把用户作为共同开发者来信任。   
-- 利用集体智慧。   
-- 通过客户的自服务来发挥长尾的力量。   
-- 软件超越单一设备。   
-- 轻量型用户界面、开发模式、和商业模式。   
  
　　今后一个公司要宣称是“Web 2.0”，就要将其特性同上述列表相测试。越符合就越名副其实。不过要记住，在某一个领域的卓越表现，可能会比对七大原则中的每个都浅尝则止，要更为有效。